CLAIMS

What is claimed is:

1. A method for resetting a partition of a multiple partition system, wherein the partition comprises a plurality of processors, the method comprising:

executing, by one processor of the plurality of processors, reset code from firmware;

building a list of reset register addresses associated with the plurality of processors;

sending an interrupt to the other processors of the plurality of processors;

resetting the other processors by writing a reset code to their associated reset registers; and

resetting the one processor by writing to its associated reset register.

2. The method of claim 1, further comprising:

storing the firmware on a read only memory.

3. The method of claim 1, further comprising:

storing the list of reset register addresses in random access memory.

4. The method of claim 1, wherein the last reset register address on the list is a reset register address for the cell that is executing the reset code.

5. The method of claim 1, further comprising:

requesting the execution of the reset code by a processor of the plurality of processors.

6. The method of claim 1, further comprising:

requesting the execution of the reset code by an operating system of the multiple partition system.

7. The method of claim 1, further comprising:

requesting the execution of the reset code by a firmware shell of the multiple partition system.

8. The method of claim 1, wherein building the list comprises:

writing the address of the one processor last in the list.

9. The method of claim 1, further comprising:

flushing a cache associated with the one processor, after sending the interrupt.

10. The method of claim 1, further comprising:

moving execution from main memory to read only memory.

11. The method of claim 1, further comprising:

switching from virtual memory mode into physical memory mode.

12. The method of claim 1, wherein the partition comprises a plurality of cells, and each cell comprises at least one processor, the method further comprises:

inventorying the plurality of cells for resetting.

13. The method of claim 1, wherein resetting the one processor occurs after resetting the other processors.

14. The method of claim 1, wherein the plurality of processors are Itanium Processor Family processors, and the firmware is system abstraction layer firmware.

15. A partition of multiple partition computer system comprising:

a plurality of processors;

firmware comprising reset code that resets a portion of the partition, wherein one processor of the plurality of processors executes the reset code; and

random access memory that is not affected by the reset code, that stores a list of addresses associated with the portion.

16. The partition of claim 15, further comprising:

read only memory that stores the firmware.

17. The partition of claim 15, further comprising:

a plurality of cells;

wherein each cell comprises at least one processor of the plurality of processors, and each cell comprises a reset register having an address that is on the list.

18. The partition of claim 17, wherein each cell is reset by writing a reset code to its associated reset register.

19. The partition of claim 17, wherein the address on the list is a reset register address for the cell that comprises the one processor.

20. The partition of claim 17, wherein each cell further comprises resources other than the at least one processor, and a portion of the resources is reset when the cell is reset.

21. The partition of claim 15, wherein the one processor is reset after the other processors of the plurality of processors are reset.

22. The partition of claim 15, wherein the plurality of processors are Itanium Processor Family processors, and the firmware is system abstraction layer firmware.

23. A computer readable medium having computer program logic recorded thereon for operating a partition of a multiple partition computer system, wherein the partition comprises a plurality of processors, the computer program logic comprising:

means for building a list of reset register addresses associated with the plurality of processors;

means for placing each processor of the plurality of processors into a known state;

means for resetting the plurality of processors by writing a reset code into their associated reset registers.